บทที่ 7

การทดสอบการทำงานของโปรแกรม

การทดสอบการทำงานของโปรแกรมจำลองซีพียู 8 บิตทั่วไปเพื่อประเมินความเหมาะสมของหลักการที่ใช้ในการพัฒนาโปรแกรมนี้ รวมทั้งเพื่อพิจารณาประสิทธิภาพของโปรแกรม ได้ทดสอบโดยการจำลองซีพียู 6502 และ 6800

ขั้นตอนหลักในการทดสอบการทำงานของโปรแกรม

1. ป้อนข้อมูลของซีพียูที่ต้องการจำลองด้วยโปรแกรม SIMEDIT
2. จำลองการทำงานของซีพียูด้วยโปรแกรม SIMMAIN

<u>การจำลองซีพียู 6502</u>

1. การป้อนข้อมูลของซีพียู 6502

   1.1 รีจิสเตอร์ 8 บิต

```
8-BIT REGISTERS
 1. A
 2. X
 3. Y
 4. P
 5.
 6.
 7.
 8.
 9.
10.
11.
12.
13.
14.
15.
```

<u>รูปที่ 7.1</u> การป้อนข้อมูลรีจิสเตอร์ของซีพียู 6502

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

1.2 แฟล็ก

```
                                        FLAGS
FLAG REGISTER NAME  P
                        Flag name       Generic flag
Flag register: bit 0  C                 CARRY
               bit 1  Z                 ZERO
               bit 2  I
               bit 3  D                 DECIMAL
               bit 4  B
               bit 5
               bit 6  V                 OVERFLOW
               bit 7  N                 SIGN
Other flags:  1
              2
              3
```

รูปที่ 7.2 การป้อนข้อมูลแฟล็กของซีพียู 6502

1.3 แอสเซมเบลอร์

```
                        ASSEMBLER AND ADDRESSING
1) Comment separator ;
2) Operand separator
3) Mnemonics determine implied addressing mode (Type "," between mnemonics.)

4) Reserved names in the operand fields (Type "," between names.)



5) Generic addressing modes:-   0 IMP
   1 ABS                5 IDXA              9 IDX1
   2 IMM                6 IDXAIMM           10 IDX1IDR
   3 IO                 7 IDXB              11 IDX2
   4 REL                8 IDXBIMM           12 IDRIDX2
                                            13 IDR
   If modes 5-16 are supposed to use, enter the register name for each of
   the selected modes.
               register name
     mode 5,6
     mode 7,8
     mode 9,10  X
     mode 11,12 Y
```

รูปที่ 7.3 การป้อนข้อมูลการทำงานของแอสเซมเบลอร์ของซีพียู 6502

1.4 สัญลักษณ์แอดเดรสชิงโหมดในภาษาแอสเซมบลี

```
                    USER-DEFINED ADDRESSING MODES

1 Pattern
          Md MNE        Mo REL        Mn OPERAND    Mb ZPG        Mw ABS
  Opcode
  BCC,BCS,BEQ,BNE,BPL,BMI,BVC,BVS
2 Pattern #
          Md IMM        Mo            Mn            Mb            Mw
  Opcode

3 Pattern ,X
          Md OPERAND    Mo            Mn            Mb ZPX        Mw ABX
  Opcode

4 Pattern ,Y
          Md OPERAND    Mo            Mn            Mb ZPY        Mw ABY
  Opcode

5 Pattern (,)X
          Md INX        Mo            Mn            Mb            Mw
  Opcode
```

```
                    USER-DEFINED ADDRESSING MODES

6 Pattern (),Y
          Md INY        Mo            Mn            Mb            Mw
  Opcode

7 Pattern ()
          Md IDR        Mo            Mn            Mb            Mw
  Opcode

8 Pattern
          Md            Mo            Mn            Mb            Mw
  Opcode

9 Pattern
          Md            Mo            Mn            Mb            Mw
  Opcode

10Pattern
          Md            Mo            Mn            Mb            Mw
  Opcode
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.4 การป้อนข้อมูลสัญลักษณ์แอดเดรสชิงโหมดในภาษาแอสเซมบลีของซีพียู6502

1.5 ชุดคำสั่ง

```
RECORD 1
     OPCODE 69              MNEMONIC ADC            MODE IMM
     NO.BYTES 2             GENERIC MODE IMM
     GENERIC INSTRUCTIONS   ADC A,IMMBYTE

     FIRST-[Home]    LAST-[End]    PREV-[PgUp]    NEXT-[PgDn]    GO TO-[^P]
       EDIT-[Enter]        DELETE-[^Y]        SAVE-[F2]        EXIT-[Esc]
```

```
RECORD 11
     OPCODE 24              MNEMONIC BIT            MODE ZPG
     NO.BYTES 2             GENERIC MODE ABS
     GENERIC INSTRUCTIONS   LOAD T1
                            AND  T1,$40
                            IF Z RESET THEN SET V
                            LOAD T1
                            AND  A,T1
                            TESTSIGN T1

     FIRST-[Home]    LAST-[End]    PREV-[PgUp]    NEXT-[PgDn]    GO TO-[^P]
       EDIT-[Enter]        DELETE-[^Y]        SAVE-[F2]        EXIT-[Esc]
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
RECORD 16
     OPCODE 50              MNEMONIC BVC            MODE REL
     NO.BYTES 2             GENERIC MODE REL
     GENERIC INSTRUCTIONS   IF V RESET THEN JUMP




    FIRST-[Home]    LAST-[End]    PREV-[PgUp]    NEXT-[PgDn]    GO TO-[^P]
      EDIT-[Enter]         DELETE-[^Y]         SAVE-[F2]         EXIT-[Esc]
```

```
RECORD 25
     OPCODE C1              MNEMONIC CMP            MODE INX
     NO.BYTES 2             GENERIC MODE IDX1IDR
     GENERIC INSTRUCTIONS   CP65 A,MEM




    FIRST-[Home]    LAST-[End]    PREV-[PgUp]    NEXT-[PgDn]    GO TO-[^P]
      EDIT-[Enter]         DELETE-[^Y]         SAVE-[F2]         EXIT-[Esc]
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.5 ตัวอย่างการป้อนข้อมูลคำสั่งของซีพียู 6502

2. การจำลองการทำงานของซีพียู 6502

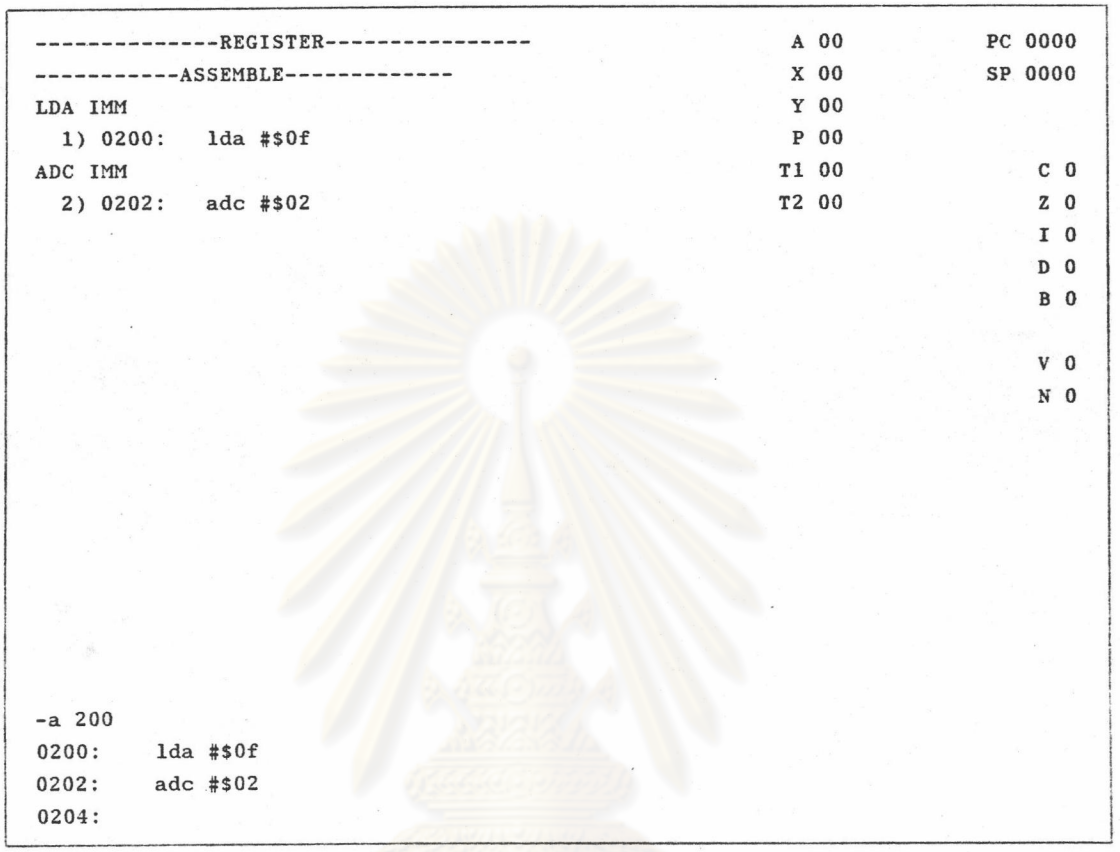
2.1 การทดสอบด้วยคำสั่ง Register

```
--------------REGISTER----------------          A 00          PC 0000
                                                  X 00          SP 0000
                                                  Y 00
                                                  P 00
                                                 T1 00                C 0
                                                 T2 00                Z 0
                                                                      I 0
                                                                      D 0
                                                                      B 0

                                                                      V 0
                                                                      N 0

READY
-r
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.6 การทดสอบการจำลองซีพียู 6502 ด้วยคำสั่ง Register

2.2 การทดสอบด้วยคำสั่ง Assemble



รูปที่ 7.7 การทดสอบการจำลองซีพียู 6502 ด้วยคำสั่ง Assemble

2.3 การทดสอบด้วยคำสั่ง Trace

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
--------------REGISTER----------------          A 0F          PC 0200
-----------ASSEMBLE-------------                 X 00          SP 0000
LDA IMM                                         Y 00
  1) 0200:   lda #$0f                           P 00
ADC IMM                                         T1 00            C 0
  2) 0202:   adc #$02                           T2 00            Z 0
-------------TRACE---------------                               I 0
                                                                 D 0
                                                                 B 0

                                                                 V 0
                                                                 N 0









0202:    adc #$02
0204:
-t
-
```

```
--------------REGISTER----------------          A 11          PC 0202
-----------ASSEMBLE-------------                 X 00          SP 0000
LDA IMM                                         Y 00
  1) 0200:   lda #$0f                           P 00
ADC IMM                                         T1 00            C 0
  2) 0202:   adc #$02                           T2 00            Z 0
-------------TRACE---------------                               I 0
-------------TRACE---------------                               D 0
                                                                 B 0

                                                                 V 0
                                                                 N 0









0204:
-t
-t
-
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.8 การทดสอบการจำลองซีพียู 6502 ด้วยคำสั่ง Trace

2.4 การทดสอบด้วยคำสั่ง Load and Assemble

```
  6)      :SUMH   EQU $310                    A 00        PC 0000
  7)      :SUML   EQU $311                    X 00        SP 0000
CLC IMP                                       Y 00
  8) 0200:        CLC                         P 00
LDA ABS                                       T1 00          C 0
  9) 0201:        LDA VAL1L                   T2 00          Z 0
ADC ABS                                                      I 0
 10) 0204:        ADC VAL2L                                  D 0
STA ABS                                                      B 0
 11) 0207:        STA SUML
LDA ABS                                                      V 0
 12) 020A:        LDA VAL1H                                  N 0
ADC ABS
 13) 020D:        ADC VAL2H
STA ABS
 14) 0210:        STA SUMH
BRK IMP
 15) 0213:        BRK
Load Completed.


READY
-r
-l tst3
-
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

<u>รูปที่ 7.9</u> การทดสอบการจำลองซีพียู 6502 ด้วยคำสั่ง Load and Assemble

2.5 การทดสอบด้วยคำสั่ง Run

```
  8) 0200:      CLC                                    A 68          PC 0213
LDA ABS                                              X 00          SP 0000
  9) 0201:      LDA VAL1L                              Y 00
ADC ABS                                              P 10
 10) 0204:      ADC VAL2L                             T1 00            C 0
STA ABS                                              T2 00            Z 0
 11) 0207:      STA SUML                                              I 0
LDA ABS                                                              D 0
 12) 020A:      LDA VAL1H                                            B 1
ADC ABS
 13) 020D:      ADC VAL2H                                            V 0
STA ABS                                                              N 0
 14) 0210:      STA SUMH
BRK IMP
 15) 0213:      BRK
Load Completed.                          0300:12 34 00 00 00 00 00 00
--------------------MEMORY--------------------  0308:56 78 00 00 00 00 00 00
---------------RUN----------------              0310:68 AC 00 00 00 00 00 00
--------------------MEMORY--------------------  0318:00 00 00 00 00 00 00 00
                                                 0320:00 00 00 00 00 00 00 00
                                                 0328:00 00 00 00 00 00 00 00
-d 300
-g 214
-d 300
-
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

<u>รูปที่ 7.10</u> การทดสอบการจำลองซีพียู 6502 ด้วยคำสั่ง Run

การจำลองซีพียู 6800

1. การป้อนข้อมูลของซีพียู 6800

1.1 รีจิสเตอร์ 8 บิต

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.11 การป้อนข้อมูลรีจิสเตอร์ 8 บิตของซีพียู 6800

1.2 รีจิสเตอร์ 16 บิต

16-BIT REGISTERS

	Word	Hbyte	Lbyte
1.	X	XH	XL
2.			
3.			
4.			
5.			
6.			
7.			
8.			
9.			
10.			
11.			
12.			
13.			
14.			
15.			

รูปที่ 7.12 การป้อนข้อมูลรีจิสเตอร์ 16 บิตของซีพียู 6800

1.3 แฟล็ก

FLAGS

FLAG REGISTER NAME F

		Flag name	Generic flag
Flag register:	bit 0	C	CARRY
	bit 1	V	OVERFLOW
	bit 2	Z	ZERO
	bit 3	N	SIGN
	bit 4	I	
	bit 5	H	HALFCARRY
	bit 6		
	bit 7		
Other flags:	1		
	2		
	3		

รูปที่ 7.13 การป้อนข้อมูลแฟล็กของซีพียู 6800

1.4 แอสเซมเบลอร์

```
                        ASSEMBLER AND ADDRESSING
1) Comment separator
2) Operand separator
3) Mnemonics determine implied addressing mode (Type "," between mnemonics.)

4) Reserved names in the operand fields (Type "," between names.)



5) Generic addressing modes:-   0 IMP
   1 ABS                5 IDXA              9 IDX1
   2 IMM                6 IDXAIMM          10 IDX1IDR
   3 IO                 7 IDXB             11 IDX2
   4 REL                8 IDXBIMM          12 IDRIDX2
                                           13 IDR
   If modes 5-16 are supposed to use, enter the register name for each of
   the selected modes.
                register name
     mode 5,6   X
     mode 7,8
     mode 9,10
     mode 11,12
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.14 การป้อนข้อมูลการทำงานของแอสเซมเบลอร์ของซีพียู 6800

1.5 สัญลักษณ์แอดเดรสชิงโหมดในภาษาแอสเซมบลี

```
                    USER-DEFINED ADDRESSING MODES

1 Pattern
          Md MNE        Mo REL        Mn OPERAND    Mb DIRECT     Mw EXTEND
  Opcode
  BRA,BCC,BCS,BEQ,BNE,BMI,BPL,BVC,BVS,BSR
2 Pattern ,X
          Md INDEX      Mo            Mn            Mb            Mw
  Opcode

3 Pattern #
          Md IMM        Mo            Mn            Mb            Mw
  Opcode

4 Pattern
          Md            Mo            Mn            Mb            Mw
  Opcode

5 Pattern
          Md            Mo            Mn            Mb            Mw
  Opcode
```

รูปที่ 7.15 การป้อนข้อมูลสัญลักษณ์แอดเดรสชิงโหมดในภาษาแอสเซมบลีของซีพียู 6800

1.6 ชุดคำสั่ง

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

```
RECORD 138
     OPCODE 7D              MNEMONIC TST            MODE EXTEND
     NO.BYTES 3             GENERIC MODE ABS
     GENERIC INSTRUCTIONS   LOAD T1
                            TESTSZ T1




    FIRST-[Home]    LAST-[End]    PREV-[PgUp]    NEXT-[PgDn]    GO TO-[^P]
      EDIT-[Enter]        DELETE-[^Y]        SAVE-[F2]        EXIT-[Esc]
```

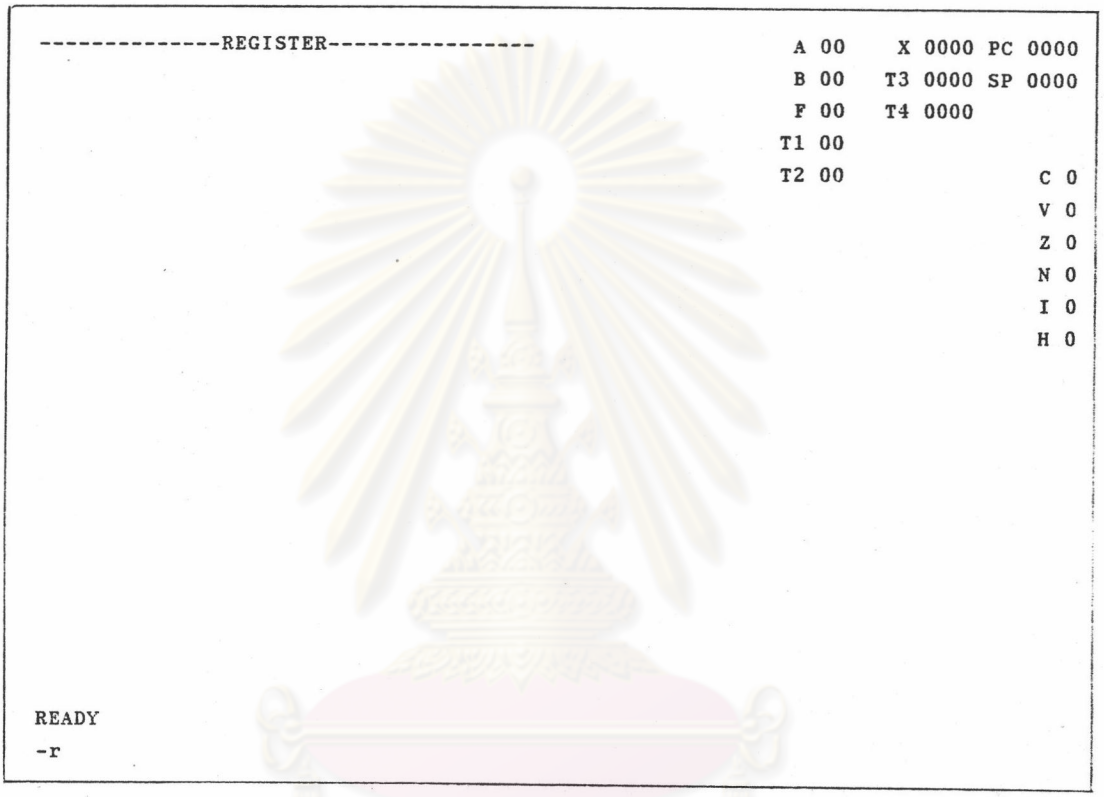
```
RECORD 142
     OPCODE 9C              MNEMONIC CPX            MODE DIRECT
     NO.BYTES 2             GENERIC MODE ABS
     GENERIC INSTRUCTIONS   LOAD T3
                            COMPARE XL,T3L
                            COMPARE XH,T3H




    FIRST-[Home]    LAST-[End]    PREV-[PgUp]    NEXT-[PgDn]    GO TO-[^P]
      EDIT-[Enter]        DELETE-[^Y]        SAVE-[F2]        EXIT-[Esc]
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.16 ตัวอย่างการป้อนข้อมูลคำสั่งของซีพียู 6800

2. การจำลองการทำงานของซีพียู 6800

2.1 การทดสอบด้วยคำสั่ง Register



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.17 การทดสอบการจำลองซีพียู 6800 ด้วยคำสั่ง Register

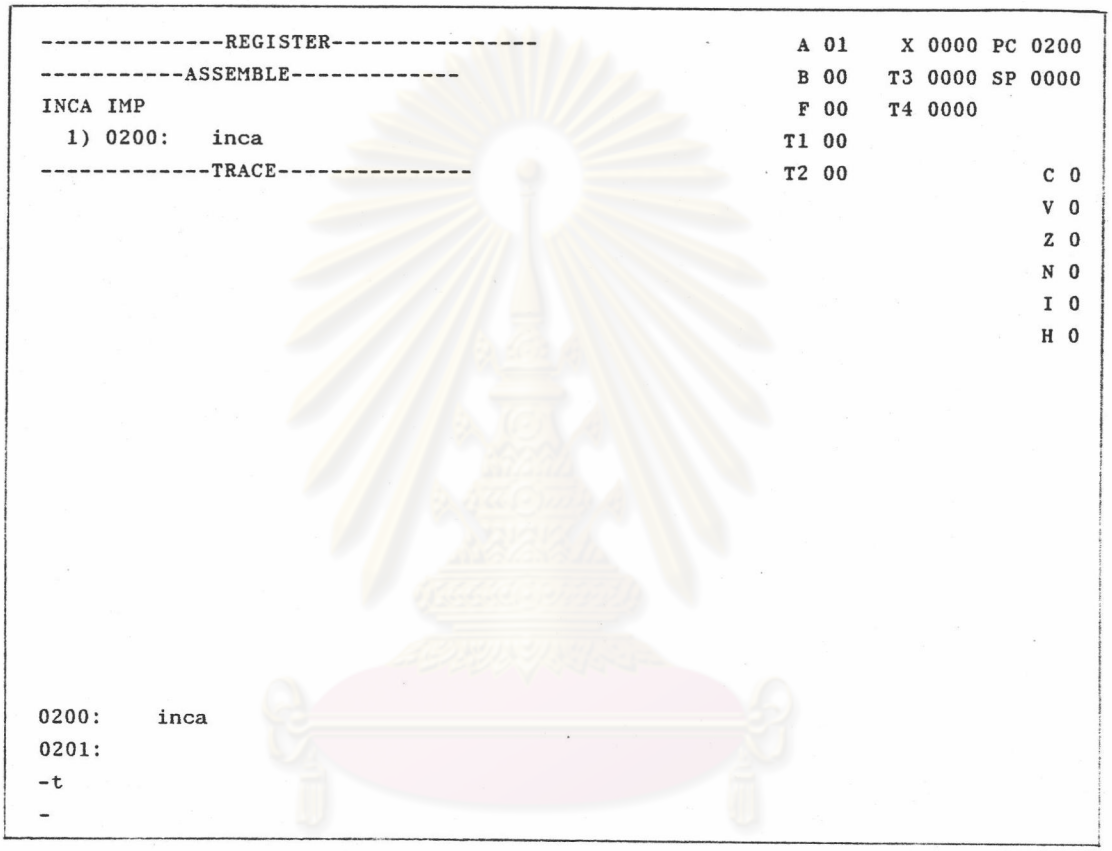
2.2 การทดสอบด้วยคำสั่ง Assemble

```
--------------REGISTER----------------          A 00    X 0000 PC 0000
-----------ASSEMBLE-------------                B 00   T3 0000 SP 0000
INCA IMP                                        F 00   T4 0000
  1) 0200:   inca                              T1 00
                                               T2 00              C 0
                                                                  V 0
                                                                  Z 0
                                                                  N 0
                                                                  I 0
                                                                  H 0
-r
-a 200
0200:    inca
0201:
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.18 การทดสอบการจำลองซีพียู 6800 ด้วยคำสั่ง Assemble

2.3 การทดสอบด้วยคำสั่ง Trace



ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.19 การทดสอบการจำลองซีพียู 6800 ด้วยคำสั่ง Trace

2.4 การทดสอบด้วยคำสั่ง Load

```
--------------REGISTER----------------                    A 17    X 0000 PC 0006
----------------LOADING-------------------                 B 00   T3 0000 SP 0000
LDAA IMM                                                   F 00   T4 0000
  1) 0000:   ldaa #$10                                    T1 00
ADDA IMM                                                   T2 00              C 0
  2) 0002:   adda #$07                                                        V 0
STAA DIRECT                                                                   Z 0
  3) 0004:   staa $08                                                         N 0
WAI IMP                                                                       I 0
  4) 0006:   wai                                                              H 0
Load Completed.
---------------RUN-----------------
--------------------MEMORY--------------------


                                                0000:00 00 00 00 00 00 00 00
                                                0008:17 00 00 00 00 00 00 00
                                                0010:00 00 00 00 00 00 00 00
                                                0018:00 00 00 00 00 00 00 00
                                                0020:00 00 00 00 00 00 00 00
                                                0028:00 00 00 00 00 00 00 00
-l test6800
-g 07
-d 00
-
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.20 การทดสอบการจำลองซีพียู 6800 ด้วยคำสั่ง Load

2.5 การทดสอบด้วยคำสั่ง Run

```
--------------REGISTER----------------          A 00    X 0000 PC 0000
----------------LOADING-------------------        B 00   T3 0000 SP 0000
LDAA IMM                                         F 00   T4 0000
  1) 0000:   ldaa #$10                          T1 00
ADDA IMM                                        T2 00              C 0
  2) 0002:   adda #$07                                             V 0
STAA DIRECT                                                        Z 0
  3) 0004:   staa $08                                              N 0
WAI IMP                                                            I 0
  4) 0006:   wai                                                   H 0
Load Completed.




READY
-r
-l test6800
-
```

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย

รูปที่ 7.21 การทดสอบการจำลองซีพียู 6800 ด้วยคำสั่ง Run

ผลการทดสอบการทำงาน

-โปรแกรมจำลองซีพียู สามารถทำงานได้ตรงตามวัตถุประสงค์ คือสามารถจำลองซีพียูได้ตามข้อมูลของซีพียูที่ป้อนให้โปรแกรม

-ประสิทธิภาพของการจำลองการทำงานในแง่ของความเร็วเป็นที่น่าพอใจ

-การจำลองซีพียู 6800 มีความไม่สมบูรณ์ของการจำลองชุดคำสั่งคือ ไม่สามารถจำลองคำสั่ง BGE,BGT,BHI,BLE,BLS,BLT ของซีพียู 6800 ได้ เนื่องจากเงื่อนไขของการกระโดดคือสถานะของแฟล็กมากกว่า 1 แฟล็ก ปัญหาที่เกิดขึ้นในกรณีนี้เป็นเพราะคำสั่ง generic แบบมีเงื่อนไขที่ออกแบบไว้ 'IF เงื่อนไข THEN คำสั่ง ' ยังไม่รองรับกรณีดังกล่าว ดังนั้นการแก้ไขก็อยู่ที่การปรับปรุงโครงสร้างข้อมูลคำสั่ง ให้มีความหลากหลายและรัดกุมยิ่งขึ้น

ศูนย์วิทยทรัพยากร
จุฬาลงกรณ์มหาวิทยาลัย